

Bridging the Gap between Mathematical Traffic Models and Operational Parameters

Kristof Sleurs, Dagang Li, Emmanuel Van Lil, and Antoine Van de Capelle

K.U.Leuven, Electrical Engineering

TELEMIC Division

Kasteelpark Arenberg 10

B-3001 Leuven, Belgium

Email: kristof.sleurs@esat.kuleuven.be

Abstract—Over the past decade and more, several accurate but complex traffic models have been developed by just as many researchers. Among the most accurate models are those based on mathematical principles that are able to model the multifractal nature of network traffic. Unfortunately these models are hardly even usable by network engineers because they lack a connection to operational parameters that can easily be estimated based on the knowledge of the network. In this paper, we try to bridge the gap between high-level parameters describing the concerning network and the inputs the modern traffic models need to generate artificial traffic. We first build a model of the behavior of TCP over the duration of a flow, and we then approximate this behavior based on simple parameters such as for example the packet transmission time and the RTT.

I. INTRODUCTION

Traffic modeling has been a domain of ongoing research for more than a decade. In these years, two main approaches towards the characterisation of network traffic have emerged: black-box modeling and structural modeling. The former treats the modeling challenge more as a theoretical, and even mathematical problem where one has to describe real, measured traffic traces as accurately as possible. The latter approach, on the other hand, uses more of a bottom-up approach to first build a virtual model of the concerning network, followed by generating - be it mathematical or via simulation - artificial traffic with characteristics that resemble real traffic as closely as possible. From an accuracy point of view, both of these approaches can be satisfying, but they both have considerable drawbacks too. As Walter Willinger and Vern Paxson state in their reply on the article by Sidney Resnick [1], the black-box techniques have the drawback of being pretty complicated and, more important, very difficult or even impossible to use based on operational parameters. They also mention that traffic modeling, performed only based on mathematical principles without making use of measurements are hardly ever useful from a network engineering point of view. Structural models [2] can offer a solution towards parsimonious modeling, but have the disadvantage to be very computationally demanding and very difficult to build when the considered network reaches a certain size. Or in other words: they are not very scalable towards modeling traffic on the backbone level.

In this paper we try to bridge the gap between mathematical, black-box models [3] [4] [5] and easy to configure, high-level

parameters. We do this by looking at the moments behavior of measured data sets, and by building a framework to explain and recreate this behavior based on high-level parameters.

It is also important to remark that, in this paper, we focus on Web traffic (WWW traffic), and more in particular the traffic flowing from Web servers towards Web clients. From our previous studies, we concluded that Web traffic takes up a significant portion of the traffic mix on the Internet nowadays (and with a high probability for a long time in the future too), and that this application is one of the main origins for the multifractal nature of aggregated traffic [6] [7].

The remainder of this paper is structured as follows: Section II first defines some important concepts and gives more information on the data sets used for this study. Next, in Section III we look at very small time scales, and derive a formula to calculate the moments at these aggregation levels. Some considerations on the parameters governing the characteristics of measured packet traces are presented in Section IV, together with a description of the TCP model we will use further on.

The following sections then handle the effects of single parameters on the global moments behavior. In the last paragraph, conclusions are drawn and future work is proposed.

II. PRELIMINARIES

A. Discrete traffic models

Discrete traffic models divide packet streams into fixed length time intervals. A bin count vector is obtained by counting the number of packets that arrived in each time interval. This bin vector can be represented by a timeseries X , with X_k representing the number of packets in bin k .

Another important concept is aggregation. An aggregated time series $X^{(m)}$ with aggregation level m is obtained by averaging the original time series X over non-overlapping blocks of m intervals:

$$X_j^{(m)} = \frac{1}{m} \sum_{k=(j-1)m+1}^{jm} X_k, \quad (1)$$

with $j = 1, 2, \dots, n/m$; where n represents the number of elements in the original time series. This principle can be applied to the bin count vector. By aggregating, packet

interarrival times are averaged over a certain block length or time scale, normally reducing peaked behavior.

To study packet streams exhibiting fractal characteristics, it does not suffice to examine the marginal distribution of the bin count vector, but it is important to look at it at different time scales. The changing behavior of this distribution over several time scales can be summarized by drawing the Variance-Time plot (VT-plot). Basically this plot describes the changing variance of the bin count vector over different aggregation levels and thus time scales. Together with the average bin contents (which does not change when aggregating according to the above definition) and possibly higher order moments, this VT-plot mathematically describes a measured packet stream, and can serve as the input to a mathematical traffic model.

B. Data sets

The results of this study are validated on several data sets. The measured data sets contain a trace which is part of the Cenic-I packet trace, publicly available online thanks to the NLANR Network Analysis Infrastructure [8]. This 48 hours contiguous data set was collected on the 10 Gigabit CENIC HPR backbone link between Sunnyvale and Los Angeles in March 2005. Three other packet streams were used for this research. The first is the Cesca-I traffic trace. This is a three hours continuous GPS-synchronized IP header trace collected at L'Anella Cientifica, the Catalan R&D network, in February 2004. The second trace is the NZIX-II data set. This collection consists of a 5-day TCP/IP header trace starting on Wednesday 5th of July, 2000. It was recorded at the New Zealand Internet Exchange, hosted by the ITS department at the University of Waikato. Thirdly, the Abilene-III data set was used. It was collected on June 1st, 2004 at the OC192c Packet-over-SONET link from Internet2's Indianapolis Abilene router node towards Kansas City. These traces are also available online on <http://pma.nlanr.net>. More information on the data sets can also be found on this website.

For the purpose of illustration, mainly results for the Cenic-I data set are included in this paper. All other measured data sets however yielded qualitatively comparable results.

III. FOTS AND VARIANCE AT SMALL TIME SCALES

The FOTS is the time scale from which the scaling behavior begins to appear. For time scales up to the FOTS, the variance and higher order moments of the bin vector can be exactly calculated. Consider a packet trace binned into a bin vector with a certain bin size. When the size of these bins is decreased, the maximal number of packets inside a bin will also decrease. Assume the bin size is set at a small enough value to obtain a maximal bin contents of one packet. In this case we can exactly calculate the variance of bin vector X , with values X_k for $k = 1..N$:

$$\begin{aligned} I_2(X) &= \frac{1}{N} \sum_{k=1}^N X_k^2 - \bar{X}^2 \\ &= \frac{P}{N} - \left(\frac{P}{N}\right)^2, \end{aligned} \quad (2)$$

with P the total number of packets in the trace. Because the correlation only appears in the VT-plot from the FOTS on, the variance will decrease with a slope of -1 versus the aggregation level on a double logarithmic scale until the FOTS is reached.

Suppose $X^{(m)}$ is the m -times aggregated bin vector with base time scale t_B . Y is the unaggregated bin vector with base time scale $t_B \cdot m$. Then the following equations hold based on the definition of an aggregated time series:

$$\begin{aligned} m \cdot X^{(m)} &= Y \\ m^2 \cdot I_2(X^{(m)}) &= I_2(Y). \end{aligned}$$

When we introduce general parameters as the packet rate R and bin size L_B , the variance of the bin vector at all time scales smaller than or equal to the FOTS can be written as:

$$I_2(Y) = R \cdot L_B. \quad (3)$$

This allows us to simply calculate the bin vector variance once the FOTS is known. From this equation, a variant can be derived for aggregated bin vectors $X^{(m)}$ with base time scale t_B :

$$I_2(X^{(m)}) = \frac{R \cdot t_B}{m}, \quad (4)$$

which indeed decreases linearly on a double logarithmic scale as expected.

Based on the last equation, we can now calculate the moments of the bin vector at the FOTS. From this equation, we can also conclude that the moments at the FOTS are linearly dependent on the packet rate and the base time scale. This warrants a normalisation of the moments behavior of a trace by these two parameters to get a fixed starting point. This normalisation will be used in the following paragraphs.

IV. MEASUREMENT ANALYSIS & TCP MODEL

At larger time scales, TCP plays an important role. Therefore we will examine the captured packet trace on a TCP flow by TCP flow manner. Specifically for our Web trace, a TCP flow generally corresponds to a single web request from a browsing Internet user. Such a TCP flow can be defined by the 4-tuple $\langle \text{Source IP, Destination IP, Source Port, Destination Port} \rangle$. To describe a single TCP flow, several parameters are necessary. First you need the starting time of the flow and the total size of the flow. The distribution of packets inside a single flow is somewhat more complicated. Based on the measurements we have to our disposal, we could conclude that the packet-train assumption [9] [10] is valid for a significant portion of all TCP flows and, more important, that these flows are the main contribution to the specific shape of the moments behavior of the concerning traffic. Following from this assumption, we need four more parameters to describe a TCP flow: Round-trip-time, packet transmission speed, maximum train size and packet loss probability.

We have built a basic TCP model that generates the timestamps of the packets of a flow based on the parameters described in the previous section. When we feed the per flow extracted values of the parameters to the model, this results

in the second and third order moments behavior depicted in Figure 1 (indicated "Original values"). The base time scale used here is $10 \mu s$. As can be seen from the graphs, the moments behavior is modeled pretty accurately, except for a minor upward deviation for aggregation levels between 10 and 10^4 . The queuing characteristics of this model were also examined, but results are omitted due to space limitations. From these results we can conclude though that the TCP model yields accurate results, and is well suited for our further study.

To obtain generally applicable results, the process of modeling TCP flow by TCP flow using the originally extracted parameters for the concerning flow has to be replaced by sampling from a general distribution, which preferably remains fixed for the entire trace. This allows us to set and vary individual parameters and examine the effect this procedure has on the global moments behaviour. For this to be possible, two main prerequisites need to be fulfilled. First, the distributions must not change with time in the scope of the trace duration we want to model. Under the assumption of stationarity, we can safely assume this condition is met. Second, the several parameters should be independent of each other, at least to such an extent warranting accurate results for our purposes. Correlation values between the different parameters indicate this condition is fulfilled too.

For this study, we prefer obtaining deterministic results when examining the effect of varying single parameters. Therefore Figure 1 also shows the moments behaviour when packet transmission time, round-trip-time, and maximal window size are fixed to one value, corresponding to the median value of the originally extracted distributions. Flow interarrival times and flow sizes are kept the same as for the original trace, and packet drops and timeouts are eliminated. The latter is done because packet dropping always has to be a random process, and introducing this will impose some variation on the moments behaviour.

V. PACKET TRANSMISSION TIME

The effects of varying the fixed packet transmission time on the normalized second order moments plot is shown in Figure 2a. The graphs show the moments behaviour for 11 different transmission time values while the RTT and maximal window size is kept constant, and no packet drops occur. From top to bottom, the curves depict the moments behaviour for transmission times ranging from 50% to 150% of the original extracted value in steps of 10%. Changing only the packet transmission time has an effect on time scales between about 100 microseconds and 100 milliseconds for the parameters used to generate this particular trace. In general, three regions can be distinguished in the moments behaviour. For very small time scales, a linear decrease on a double logarithmic scale exists. At larger time scales, two regions are separated by a break-point (which is located at an aggregation level of about 10^4 in Figure 2a). As will become clear in the following paragraphs, these three regions are bounded by the transmission time and the round-trip-time. In this section, we focus on the position of the first endpoint (at an aggregation

level of about 10^1 in Figure 2a) and the shape of the moments behaviour between the two endpoints.

Figure 2a shows that the first endpoint, which will be further designated as m_{tt} , shifts towards higher aggregation levels when the transmission time is increased. m_{tt} is linearly dependent of the transmission time, following the relationship:

$$m_{tt} = \frac{\text{packet transmission time}}{t_B}. \quad (5)$$

As the moments behaviour is linear between the FOTS and the first endpoint for traffic generated by our TCP model, we can quite easily extract the variance and higher order moments of the bin vector at the time scale corresponding to the endpoint. In Section III, we derived a formula for the moments at time scales below the FOTS. Using equation (4), we can compute the normalized variance at m_{tt} :

$$I_{2,norm}(X^{(m_{tt})}) = \frac{t_B^2}{\text{packet transmission time}} \quad (6)$$

When we look at the shape of the moments behaviour between the endpoints, it can be quite accurately approximated with a rational function. As the graphs for all transmission time values coincide when they near the second endpoint, and they seem to have a linear asymptote with slope respectively equal to -1 , -2 and -3 on a double logarithmic scale for the different moments, the following function can be used for the variance:

$$I_{2,norm,approx}(X^{(m)}) = \frac{1}{A_{I_{2,tt}} \cdot m + B_{I_{2,tt}}} \quad (7)$$

The functions for the higher order moments are completely similar. To get the unknown coefficients, we need two points on every graph. One point is already defined as the first endpoint. The second point will be defined by the round-trip-time and maximal TCP window size. Figure 2b shows the moments behaviour (solid line) and the approximating function (dash-dotted line) for a transmission time of $1.242 \cdot 10^{-4} s$, which equals the extracted transmission time for the Cenix-I packet trace. From the graphs it is clear that the approximating function decreases slightly faster than the original moments.

VI. TCP-FLOW MAXIMAL WINDOW SIZE

To avoid flooding the receiving host and congesting the network, TCP flows are not allowed to grow in size indefinitely. The former is taken care of by the receiver itself, by means of the *AdvertisedWindow*. In practice, this means that the receiving host informs the sending host about the maximal amount of bytes that can be "in flight" for the flow in question. A packet is denoted as being in flight when it has been sent by the sender, but not yet acknowledged by the receiver. This mechanism directly imposes a maximum limit on the resulting TCP window, as used in the sending procedure.

Another, more indirect way of setting a maximal window size is achieved with the congestion control mechanism of TCP. In practise, the resulting window will fluctuate about some semi-fixed value, which is governed by the fairness principle of TCP

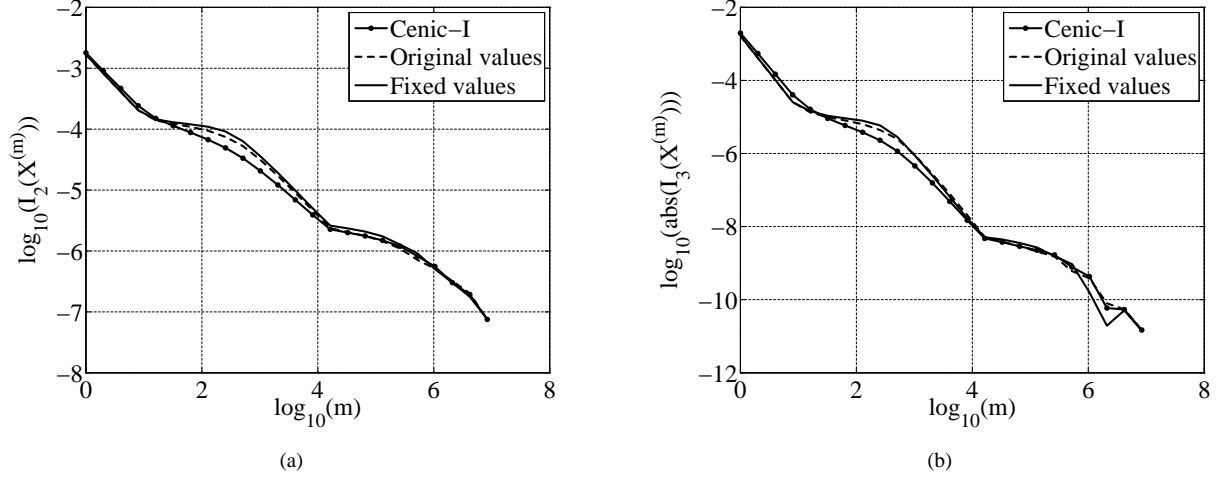


Fig. 1: Accuracy of the basic TCP model in approximating the moments behavior of the original packet trace. A distinction was made between using per flow extracted values and using a single, fixed value for the parameters. (a) Variance, (b) third order semi-invariant moment.

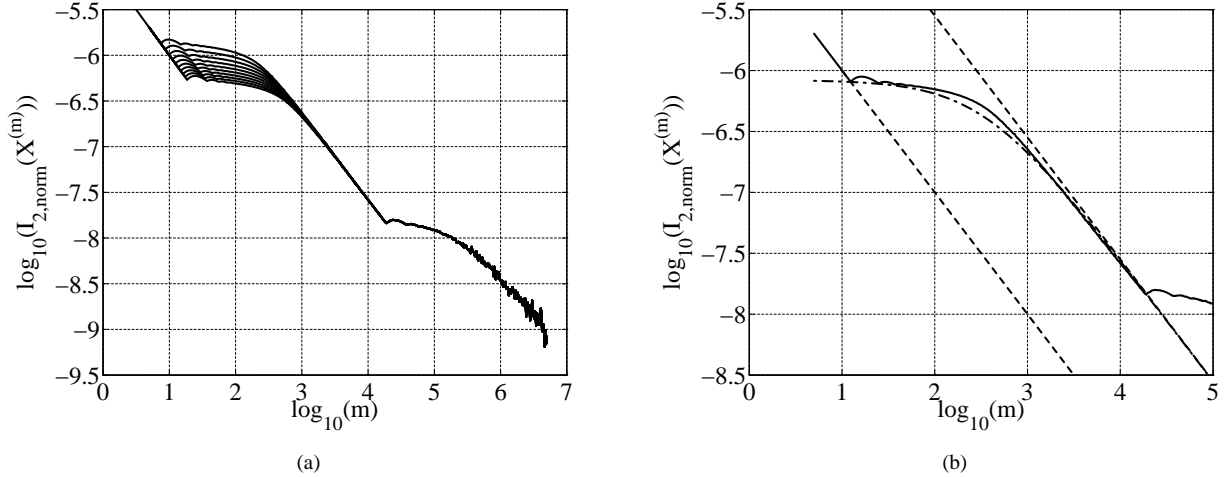


Fig. 2: Effect of varying the transmission time on the moments behaviour. (a) Normalized variance versus aggregation level, (b) shape of the second order moment for aggregation levels above m_{tt} .

and the global level of congestion in the network. It is semi-fixed because it can change in time due to load fluctuations in the network.

We previously mentioned that the maximal window size was set to a fixed value equalling the median of the extracted per-flow maximal window size. All packet drops were also eliminated. This effectively means that we eliminated one of the external influences on the resulting TCP window size, namely the packet dropping mechanism. This results in the disappearance of variation on this window size outside the period where the flow is probing for the network capacity (i.e. the Slow Start phase). We thus approximated the combined behaviour resulting from both window limiting algorithms (the *AdvertisedWindow* and the *CongestionWindow*) by setting the maximal TCP window to the median of all extracted values.

Therefore, this value does not directly correspond to the value of the *AdvertisedWindow*. We will make abstraction of the reason for the maximal window size for now, and evaluate the influence of this parameter on the moments behaviour.

Figure 3a shows the moments behaviour when the maximal window size is varied while keeping the packet transmission time and the round trip time fixed and eliminating all packet drops. It is clear that the maximal window size does not influence the aggregation levels at which the two endpoints occur. Basically, varying the maximal window size shifts the second endpoint up and down. In the next section, we will see that the round trip time governs the aggregation level at which the second endpoint occurs. Combined with the maximal window size, the position of the second endpoint can thus be completely defined.

To derive a method to predict the moments at the second endpoint, we have extracted several (maximal window size, variance at m_{RTT}) pairs, and plotted them in Figure 3b. Apparently the variance increases monotonically until a certain limit is reached. Intuitively, this limit could be predicted as the maximal flow size is limited, and therefore a threshold has to exist above which increasing the maximal window size any further has no effect. Important to note is that the underlying reason for the shape of these curves is the way file sizes are distributed. We will use this to derive an approximation for the measured behaviour. To build up this approximation, we will start from the fact that when the time scale corresponds to the round trip time, the bin vector basically consists of the subsequent packet trains. When we assume that the different TCP-flows are approximately independent of each other, we can sum the contribution to the moments of the individual flows and thus get an approximation of the global moment values.

We will illustrate this procedure for the second order moment. For a certain maximal window size, we need to sum the total variances of all flows in the trace. The variance of a single flow equals the squared deviation from the average train size, summed over all trains and divided by the amount of train "slots" in the trace. When summed over all flows, we call this value the sum of the squared deviations (SSD).

Now the variance value at m_{RTT} equals:

$$\log_{10} \left(I_{2,norm} \left(X^{(m_{RTT})} \right) \right) = \log_{10} \left(\frac{SSD \cdot t_B^2}{P \cdot RTT} \right), \quad (8)$$

using Equation (9). This function is shown in Figure 3b as a solid line, and approximates the measured values pretty well. To use this model in practise, the SSD values for a range of maximal window sizes have to be calculated for a certain flow size distribution. As these values only depend on the flow size distribution and the total number of packets, this only has to be done once. Moreover, as the size of Web Requests is more or less stable over the Internet, a fixed table can be used as an approximation for all traces containing web requests to a random subset of all web pages on the Internet [11].

Once the value of SSD is known, the above formula holds, independent of the packet transmission time and the round trip time. Therefore the position of the second endpoint can be easily calculated.

VII. TCP-FLOW ROUND TRIP TIME

It became clear in the previous paragraphs that the round trip time has an effect on time scales beyond the second endpoint, as well as on the position of this second endpoint. These effects can be clearly seen in Figure 4a. The round trip time defines the time scale at which the second endpoint (m_{RTT}) occurs, according to the relationship:

$$m_{RTT} = \frac{RTT}{t_B}. \quad (9)$$

The moments behaviour beyond the second endpoint is very similar to that between the endpoints, caused by the packet

transmission time. At high aggregation levels, the variance as well as the higher order moments decrease linearly on a double logarithmic scale, indicating that subsequent contents of the bin vector are independent of each other. We will first derive an approximation for the intercept of this linear behaviour. Then we will take a look at the shape of the curve between m_{RTT} and the linear part.

To estimate the position of the linear variance behaviour, we start from the observation that, when the time scale is big enough, the bin vector is basically a summation of the sizes of all flows that start within a certain time interval. More specifically, the time scale has to be at least an order of magnitude bigger than the largest flow size. We capped the flow sizes at 1000 packets, which corresponds to a flow duration of about 7 seconds for the extracted parameter values and no packet drops. This duration is of course dependent on the RTT, the maximal TCP window size and the drop probability. Therefore, to be on the safe side, we use a time scale of 1000 seconds, which corresponds to an aggregation level $m_{1000} = 1000/t_B$.

As mentioned in [12], the Web Request interarrival time distribution can be well modeled by an exponential distribution. This results in a Poisson distributed number of flows starting in a time interval of 1000 seconds. We generate an artificial bin vector at the concerning time scale by repeating the following procedure for all elements: sample from the Poisson distribution, assign to each flow a random flow size from the extracted flow size distribution, and add all of these sizes up to get the total packet count. We then calculate the variance of this bin vector. For this value to be independent of the flow arrival rate and the time scale, it has to be normalised by the used values of these parameters. We then get a normalised variance value (further on denoted as NVV) that remains valid as long as the flow size distribution is not altered.

Based on this value, the variance at m_{1000} can be calculated:

$$\log_{10} \left(I_{2,norm} \left(X^{(m_{1000})} \right) \right) = \log_{10} \left(NVV \cdot \frac{F}{P} \cdot \frac{t_B}{m_{1000}} \right). \quad (10)$$

Here, F is the total number of flows and P is the total number of packets. As the average amount of packets per flow is typical for the flow size distribution, we can also include this in a normalised version of NVV. The value of this variable for the Cenix-I data set is 325.

Considering the fact that the VT-plot at the concerning time scale is a straight line with slope -1 on a double logarithmic scale, we can extract the intercept of this line as follows:

$$\log_{10} \left(I_{2,norm,linear} \left(X^{(1)} \right) \right) = \log_{10} (NVV_{norm} \cdot t_B). \quad (11)$$

The procedure for the higher order moments is completely similar, only instead of the variance the appropriate moment of the bin vector has to be calculated. After normalisation, the intercept of the line approximating the higher order moment behaviour for large time scales can also be found by

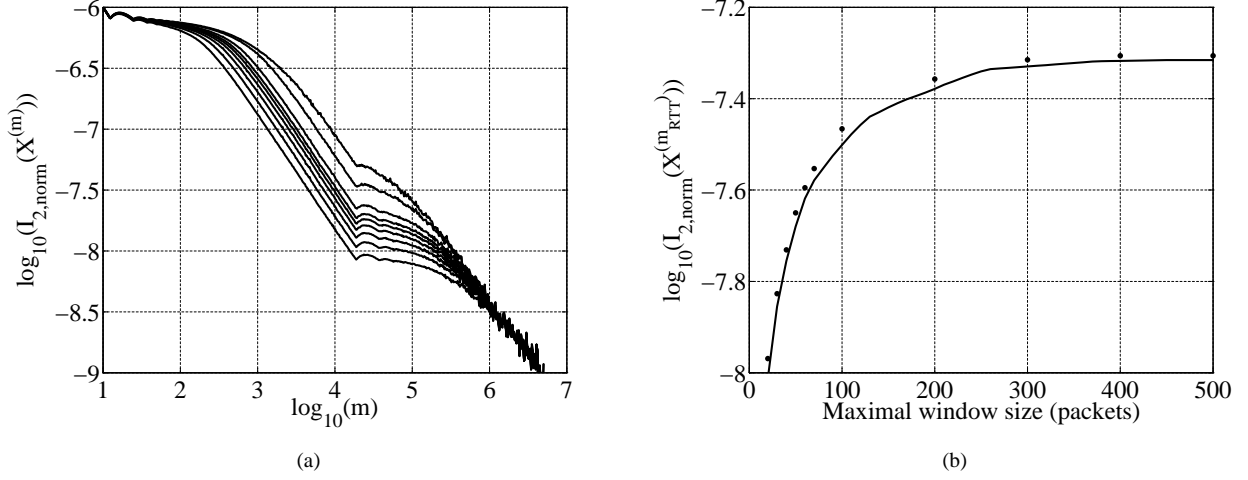


Fig. 3: Effect of varying the maximal window size on the moments behaviour. (a) Normalized variance versus aggregation level, (b) vertical position of the second endpoint versus maximal window size for the second order moment.

multiplication with the base time scale.

Starting from the position of the second endpoint and the intercept of the linear behaviour at large time scales, we now try to approximate the shape of the moments behaviour for time scales beyond the second endpoint. We will use the first part of a Sigmoid function for this approximation, namely the hyperbolic tangent. The specific function that we will use is:

$$\begin{aligned} \log_{10} \left(I_{2, \text{norm}, \text{approx}}(X^{(m)}) \right) \\ = -\tanh(A_{I_2, \text{RTT}}(\log_{10}(m) - B_{I_2, \text{RTT}}))^3 \\ + (\log_{10}(m) - B_{I_2, \text{RTT}}) - C_{I_2, \text{RTT}}. \end{aligned} \quad (12)$$

This function was chosen to provide three degrees of freedom: A defines how fast the curve approximates the oblique asymptote, and B and C set respectively the horizontal and vertical position of the curve. The bigger A is, the less gentle the curve is. For the different order moments, we multiply the hyperbolic tangent with a factor, resulting in a change in slope of the oblique asymptote.

The function parameters can be calculated as follows:

$$C_{I_M, \text{RTT}} = (M - 1) - \log_{10}(I_{2, \text{norm}}(X^{(m_{\text{RTT}})})) \quad (13)$$

$$B_{I_M, \text{RTT}} = \frac{1}{M - 1} (\log_{10}(I_{M, \text{norm}, \text{linear}}(X^{(1)})) + C_{I_M, \text{RTT}}) \quad (14)$$

$$A_{I_M, \text{RTT}} = 0.15. \quad (15)$$

Thus, C and B can be calculated from the values obtained in the previous section and earlier in this section. The measured values were fitted optimally when A is set to 0.15. For the second order moments of the Cenit-I data set, this approximation is shown in Figure 4b. For this plot, the original values for all extracted parameters were used, except for the packet drops, which were all eliminated. A very good fit of the moments behaviour can be achieved in all cases.

VIII. CONCLUSION

In this paper, we developed a method of approximating the mathematical behavior of a packet stream, captured on the Internet, using only high-level parameters. We started from a TCP model that mimics the behavior of real traffic quite well, followed by a parameter by parameter study of the inputs to this model. In this paper, we treated the packet transmission time, the round trip time and the maximal TCP window size. Due to space limitations we did not include packet loss and flow interarrival time and size distribution yet.

To conclude the paper, we give an indication of the accuracy of the approximation in Figure 5. The concerning data set was not used in the modeling process, and is selected at random from the Abilene-III data set. It is clear that the behavior of the TCP model and the approximation nearly coincide. We can thus state that our approximation fits the model pretty well. When compared to the measured data set, the fit is again quite accurate, except for a minor upward deviation at intermediate time scales. This deviation is basically due to fluctuations in the packet transmission time within a packet train. Unfortunately, this issue could not yet be resolved without overly complicating our model, and is thus left as future work.

Finally, we remark again that all studies presented in this paper took the second order moments as well as the third and the fourth order moments into account. All results were qualitatively comparable. As is explained in [13], next to the variance, especially the third order moment has to be considered to achieve an accurate modeling of real network traffic. We also need to note that the packet-train assumption might not be valid in all scenarios. When the bandwidth and the round trip time of the network is completely different from these found in the data sets we used, packet trains might not be the main cause of the particular moments behaviour of the trace. This could also be a topic of future work.

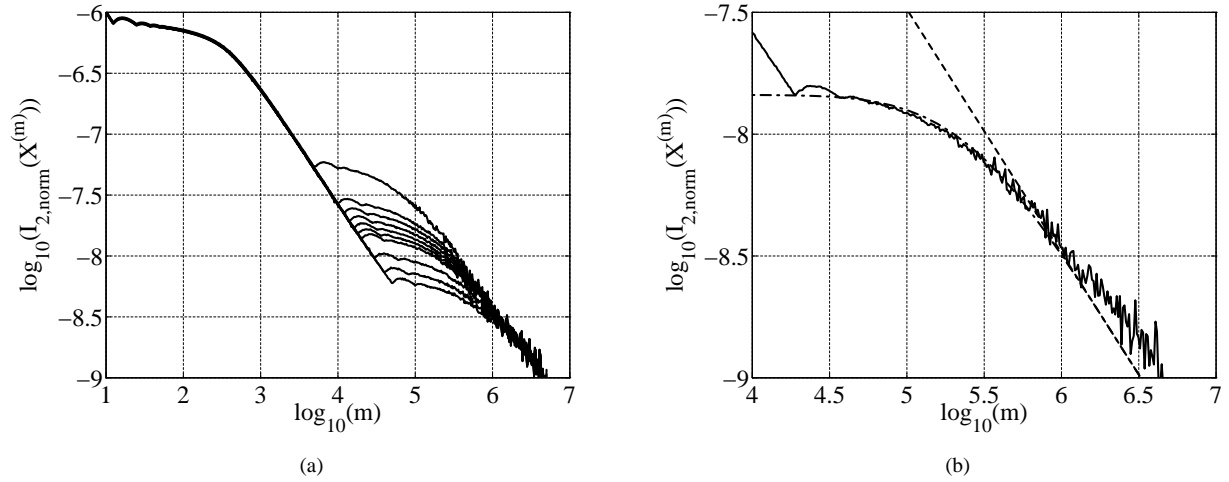


Fig. 4: Effect of varying the RTT on the moments behaviour. (a) Normalized variance versus aggregation level, (b) shape of the second order moment for aggregation levels beyond m_{RTT} .

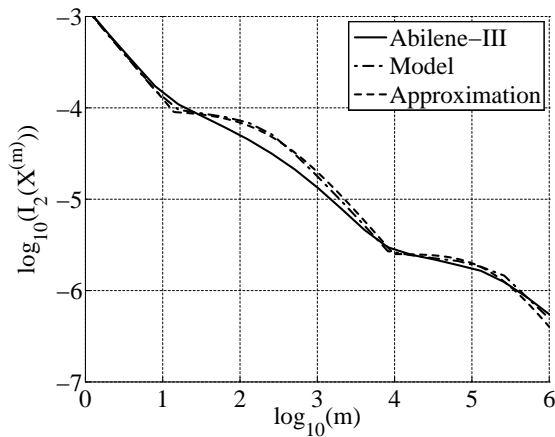


Fig. 5: Comparison of the moments behaviour of the measured packet trace, the TCP model and the approximation.

ACKNOWLEDGMENT

Kristof Sleurs is a research assistant of the Fund for Scientific Research Flanders (FWO-Vlaanderen), Belgium.

REFERENCES

- [1] S. Resnick, "Heavy tail modeling and teletraffic data," *The Annals of Statistics*, vol. 25, no. 5, pp. 1805–1869, 1997.
- [2] Hyoungh-Kee Choi and John O. Limb, "A Behavioral Model of Web Traffic," in *ICNP '99: Proceedings of the Seventh Annual International Conference on Network Protocols*. Washington, DC, USA: IEEE Computer Society, 1999, p. 327.
- [3] J. Potemans, J. Theunis, P. Leys, B. Van den Broeck, E. Van Lil, and A. Van de Capelle, "Advanced Traffic Modeling: Fitting Third Order Moments," in *Proc. to IEEE Global Conference on Communications (GLOBECOM '03)*. San Francisco, CA, USA: IEEE Global Telecommunications Conference, December 2003.
- [4] A. Gilbert, W. Willinger, and A. Feldmann, "Scaling analysis of conservative cascades, with applications to network traffic," *IEEE Transactions on Information Theory*, vol. 45, no. 3, pp. 971–991, April 1999.
- [5] A. Feldmann, A. Gilbert, and W. Willinger, "Data networks as cascades: investigating the multifractal nature of Internet WAN traffic," in *Proceedings of the ACM SIGCOMM '98 conference on Applications, technologies, architectures, and protocols for computer communication*. Vancouver, British Columbia, Canada: ACM Press, New York, NY, USA, 1998, pp. 42–55.
- [6] M. S. Taqqu, V. Teverovsky, and W. Willinger, "Is network traffic self-similar or multifractal?" *Fractals*, vol. 5, pp. 63–73, 1997.
- [7] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz, "The changing nature of network traffic: scaling phenomena," *ACM SIGCOMM Computer Communication Review*, vol. 28, no. 2, pp. 5–29, April 1998.
- [8] A. McGregor, H. W. Braun, and J. Brown, "The NLANR Network Analysis Infrastructure," *IEEE Communications Magazine*, vol. 38, no. 5, pp. 122–128, May 2000. [Online]. Available: <http://pma.nlanr.net>
- [9] Srinivas Shakkottai, R. Srikant, Nevil Brownlee, Andre Broido, and Kc Claffy, "The RTT distribution of TCP flows in the Internet and its impact on TCP-based flow control." Technical report, Cooperative Association for Internet Data Analysis (CAIDA), 2004.
- [10] Shriram Sarvotham, Rudolf Riedi, and Richard Baraniuk, "Connection-level analysis and modeling of network traffic," in *IMW '01: Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*. San Francisco, California, USA: ACM Press, New York, NY, USA, 2001, pp. 99–103.
- [11] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 835–846, December 1997.
- [12] Kihong Park, Gitae Kim, and Mark Crovella, "On the relationship between file sizes, transport protocols, and self-similar network traffic," in *ICNP '96: Proceedings of the 1996 International Conference on Network Protocols (ICNP '96)*. Washington, DC, USA: IEEE Computer Society, 1996, p. 171.
- [13] J. Potemans, "Mathematical Modelling of Aggregated Traffic Streams on the Internet," Ph.D. dissertation, K.U.Leuven, Leuven, Belgium, April 2007.